Reliability Engineering and System Safety 142 (2015) 123–133

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

Reliability analysis of large phased-mission systems with repairable components based on success-state sampling

Ji-Min Lu[a,b,*], Xiao-Yue Wu[a], Yiliu Liu[b], Mary Ann Lundteigen[b]

[a] College of Information System and Management, National University of Defense Technology, Changsha, Hunan 410073, China
[b] Department of Production and Quality Engineering, Norwegian University of Science and Technology, Trondheim, Norway

ARTICLE INFO

Article history:
Received 13 October 2014
Received in revised form
17 April 2015
Accepted 15 May 2015
Available online 27 May 2015

Keywords:
Phased mission systems
Reliability assessment
Repairable components
Binary decision diagram
Sampling

ABSTRACT

In many engineering applications, some phased-mission systems (PMS) may contain a large number of phases and repairable components. Traditional binary decision diagram (BDD) based methods or state-enumeration methods can suffer from the BDD explosion or the state explosion for this kind of PMS. This paper presents a non-simulation method for the reliability analysis of large PMS. In our approach, the system reliability is approximated by the system availability at discrete time. The discrete-time availability is modeled by the sampling of success states, which avoids the BDD explosion as the number of phases increases. Furthermore, BDDs are used to simplify success states, and enable our model to avoid the state-explosion problem. Two real-world PMS are analyzed to illustrate that the time and the storage cost of our approach do not increase exponentially with the number of components and phases in the PMS.

1. Introduction

In some real-world applications, a mission is usually accomplished by different equipments during different periods of time. Phased-mission systems (PMS) are commonly used to describe the system where the component stress and the system configuration may vary from phase to phase. A typical example of PMS is the railway autopilot system which includes many accelerating and decelerating phases. In PMS, some components, such as engines, may be used many times during different phases, and become idle in specific phases. This phenomenon makes the reliability analysis of PMS more complicated than that of single-phase systems. Real-world PMS may be the combination of many complex situations such as the repairable components, changing success criteria, flexible duration of phases.

Extensive research efforts have been expended in the reliability assessment of PMS since 1970s. Generally, existing methodologies can be categorized into the simulation methods [1–3] and the analytical methods. The analytical approaches can be further classified into the state-enumeration methods, the combinatorial methods, and the modular methods which combines the former two approaches. The state-enumeration methods [4–8] are mainly used to analyze PMS with repairable components. It is well known that the state-enumeration methods may suffer from the state-explosion problem, which makes these methods inapplicable to the PMS with many components. However, the state-enumeration methods have the advantage in analyzing the PMS with many phases. Conversely, the combinatorial approaches [9–18], especially the binary-decision-diagram (BDD) based methods [11–18], are efficient in analyzing the PMS with many components. Nevertheless, the BDD-based methods for PMS analysis require that the cross-phase BDD must be evaluated from the first phase to the last phase. This requirement inevitably leads to the explosion in BDD nodes (or BDD paths) when the PMS contains a large number of phases.

The modular methods [19–22] are designed with an aim to capture the advantages of both the state-enumeration methods and the combinatorial methods. An example of the modular methods is the "BDD and Markov" method [19,20] which is applicable to PMS with a lot of repairable components. However, this method contains an impractical assumption that repaired components cannot be reused immediately after repair. Similar to the combinatorial methods, this "BDD and Markov" approach faces the exponential growth in execution time (and in memory requirement) as the number of phases increases.

In order to analyze large systems, considerable efforts have been made in the study of truncation and compression. One common strategy [23–27] is to apply truncation to BDD (or fault tree) to slow down the exponential growth of time cost or space cost. Another strategy [28,29] focuses on the shrinkage of the Markovian methods through the compression storage of large sparse matrices. However,

* Corresponding author at: College of Information System and Management, National University of Defense Technology, Changsha, Hunan 410073, China.
E-mail address: jeemanlv@gmail.com (J.-M. Lu).

http://dx.doi.org/10.1016/j.ress.2015.05.010
0951-8320/

some experimental works [30–33] show that the fixed truncation limit may result in the significant error in results. When the flexible truncation limit is applied to the PMS analysis, the time (or the storage) cost can be diminished. However, the truncation benefit will quickly disappear as the size of PMS increases.

Literature instances [34–36] show that the BDD-based methods are efficient for PMS with many components, while the state-enumeration methods work efficiently for PMS with many phases. In order to analyze large PMS with repairable components, this paper proposes the "simplified success states" which are generated by BDD, and are evaluated by state mappings. Compared to the traditional BDD and Markov method [19,20], the proposed method not only eliminates the impractical assumption, but also avoids the BDD explosion problem (exponential growth of storage cost or time cost) when the number of phases increases.

The remainder of the paper is organized as follows. Section 2 describes the proposed method, and analyzes the discretization error and the computational time. Section 3 illustrates our method with the airliner-flight mission and the satellite-communication mission. Three kinds of methods (BDD and Markov, Petri-net simulation, and our approach) are compared to show the efficiency of the proposed algorithm. Lastly, Section 4 summarizes the advantages and gives directions for future work.

2. Sampling model for reliability analysis

The proposed method in this section is based on the assumption that the life and the repair time of components are independent variables of exponential distributions. The first step of the algorithm is to find "simplified success states" for each phase. With simplified success states, the model avoids the state-explosion problem. In the second step, we introduce the "discrete-time availability", and evaluate it through simplified success states. This evaluation method enables the algorithm to consider repairable components without the assumption in [19,20]. Throughout the paper, the phase requirement of PMS is phase-OR, i.e., the mission is assumed to fail if the system fails during any phase.

2.1. Simplified success states based on BDD

Success states, as well as failure states, are widely used in many state-enumeration methods. The success state $S(t)=(a_1,a_2,...,a_n)$ refers to the combination of components' states which renders the system succeeds at time t. The element a_K is $u_K(t)$ if the state of the component K is up (at time t) and $d_K(t)$ otherwise. Because of the state-explosion problem, the number of $S(t)$ can be huge even for a small system. In this section, we introduce a symbol $e_K(t)$ to reduce the number of $S(t)$. $e_K(t)$ is of the form

$$e_K(t)=\begin{cases} u_K(t), \text{if the state of K is up at time } t \\ d_K(t), \text{if the state of K is down at time } t \end{cases} \tag{1}$$

Using $e_K(t)$, we combine many $S(t)$ together to form a simplified $S(t)$. For instance consider the system whose BDD is shown in Fig. 1. The simplified $S(t)$ are equivalent to the paths from the top node to the bottom node 1 in the BDD. Therefore, the number of simplified $S(t)$ equals to the number of BDD paths, and it is much smaller than the number of traditional $S(t)$.

The generation of simplified $S(t)$ can be programmed through the mature algorithms of BDD (see [37,38] for BDD generation). With an efficient ordering strategy, the number of BDD paths (and the number of simplified $S(t)$) does not increase exponentially with the number of components. Before assessing the PMS reliability, we need to obtain simplified $S(t)$ for each phase.

2.2. Reliability evaluation based on simplified success states

2.2.1. Single-phase systems with independent component repairs

According to the definition of reliability, we note that the system is reliable before time t indicates that the system is always available before t. In the following, we approximate the system reliability by the system availability at discrete time points. Consider the inequality (2) which depicts the difference between the system reliability $R_{sys}(t)$ and the system availability $A_{sys}(t)$.

$$\begin{aligned} R_{sys}(t) &= \Pr\{\text{non} - \text{repairable system succeeds at time } [0,\ t]\} \\ &\le \Pr\{\text{repairable system succeeds at discrete time } 0, 0.1, 0.2, ..., t\} \\ &\le \Pr\{\text{repairable system succeeds at discrete time } 0, 1, 2, ..., t\} \\ &\le \Pr\{\text{repairable system succeeds at discrete time } 0 \text{ and } t\} = A_{sys}(t) \end{aligned} \tag{2}$$

Given a group of discrete time points $(\tau_1,\tau_2,...,t)$, we call Eq. (3) the "discrete-time availability $\hat{A}_{sys}$" with regard to $(\tau_1,\tau_2,...,t)$. $\hat{A}_{sys}=R_{sys}(t)$ if the system contains no repairable components.

$$\hat{A}_{sys}(\tau_1,\tau_2,...,t)=\Pr\{\text{repairable system succeeds at discrete time } 0,\tau_1,\tau_2,...,t\} \tag{3}$$

Consider the system whose structure is shown in Fig. 2. In order to evaluate $R_{sys}(t)$ at $t=3$, we first consider time as discrete, and find the simplified $S(t)$ at time 0, 1, 2, and 3, as shown in Fig. 2. From the definition of the discrete-time availability, we can see that the probability of $S(t)$ must come from the mapping $S(t-1)\rightarrow S(t)$. Hence,

$$\begin{aligned} R_{sys}(t) &\approx \Pr(\text{repairable system succeeds at time } 0,1,2,t) \\ &= \Pr(\text{SysState} \approx \left\{S(1)_j\right\} \text{ at } t=1; ...; \text{SysState} \in \left\{S(t)_j\right\} \text{ at time } t) \\ &= \sum\nolimits_j \Pr\left\{S(t)_j\right\} \end{aligned} \tag{4}$$

Eq. (4) provides a method to assess $R_{sys}(t)$ through $\Pr\{S(t)\}$. $\Pr\{S(t)\}$ are evaluated in chronological order. At the start of the mission, all components are assumed to be initially operational. Therefore, there is only one simplified $S(t)$ at $t=0$, that is, $S(0)=(u_A,u_B,u_C,u_D)$. The probability of $S(0)$ is

$$\Pr\{S(0)\}=R_{sys}(0)=1 \tag{5}$$

Next, we evaluate the probabilities of simplified $S(t)$ at $t=1$ through state mappings. Take $S(1)_1=(e_A,e_B,e_C,u_D)$ (first simplified $S(t)$ at $t=1$) for instance, its probability is given by

$$\begin{aligned} \Pr\{S(1)_1\} &= \Pr\{S(0)\}\cdot\Pr\{S(0)\rightarrow S(1)_1\} \\ &= 1\cdot\Pr\{u_A(0)\rightarrow e_A(1)\}\cdot\Pr\{u_B(0)\rightarrow e_B(1)\}\cdot\Pr\{u_C(0)\rightarrow e_C(1)\}\cdot\Pr\{u_D(0)\rightarrow u_D(1)\} \end{aligned} \tag{6}$$

Since $e_K(1)$ is either $u_K(1)$ or $d_K(1)$, we have

$$\begin{aligned} &\Pr\{u_K(t_0)\rightarrow e_K(t_1)\}=1 \\ &\Pr\{d_K(t_0)\rightarrow e_K(t_1)\}=1 \quad (t_0<t_1) \\ &\Pr\{e_K(t_0)\rightarrow e_K(t_1)\}=1 \end{aligned} \tag{7}$$

Hence, Eq. (6) can be rewritten as:

$$\begin{aligned} \Pr\{S(1)_1\} &= \Pr\{u_D(0)\rightarrow u_D(1)\} \\ &= \text{UpState}_K\cdot\exp(\begin{pmatrix} -\lambda_K & \lambda_K \\ \mu_K & -\mu_K \end{pmatrix}\cdot 1)\cdot\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}\cdot\begin{pmatrix} 1 \\ 1 \end{pmatrix} \end{aligned} \tag{8}$$

where $\text{UpState}_K=(1,0)$ represents K is operational at $t=0$. The matrix and the vector $[1,0;0,0]\cdot[1,1]'$ are used to extract the first element from $\text{UpState}_K\cdot\exp([-\lambda,\lambda;\mu,-\mu]\cdot 1)$. Eq. (8) holds under the assumption that the life and the repair time of components follow exponential distributions.